

## **BAB II**

### **TINJAUAN PUSTAKA DAN DASAR TEORI**

#### **2. 1. Tinjauan Pustaka**

Sebelumnya aplikasi chat messenger sudah pernah dibuat dengan mengangkat kasus dan penggunaan teknologi yang berbeda-beda, pada table 2.1 menunjukkan penelitian-penelitian yang sudah pernah dilakukan oleh beberapa penulis.

Pada skripsi yang ditulis oleh Andre yang berisi mengenai membangun Aplikasi panduan kepramukaan berbasis android menggunakan format pertukaran data JSON (Java Script Object Notation).

Pada skripsi yang ditulis oleh Bakhtiar Wijayanto yang berisikan tentang membangun aplikasi chat berbasis android. Pada skripsi ini memanfaatkan teknologi HTML, CSS, PHP. Pada penelitian ini menonjolkan kemudahan dalam login, serta dapat bertukar informasi berupa file .

Pada skripsi yang ditulis oleh Fitri Hardianto yang berisi mengenai cara membangun aplikasi grupchat yang realtime menggunakan websocket. Pada penelitian tersebut membahas mengenai aplikasi grupchat yang hanya dapat bertukar informasi berupa teks berbasis web.

Pada skripsi yang ditulis oleh Nicholas dengan berisi mengenai bagaimana membuat chatroom menggunakan AJAX dan PHP berbasis

web, yang juga membahas mengenai aplikasi chat berbasis web yang hanya dapat bertukar informasi berupa teks.

Kemudian pada skripsi yang ditulis oleh Fahsyani Ade Putra Wally yang berisi mengenai pembuatan aplikasi chat untuk menyatukan berbagai aplikasi chat dalam satu aplikasi. aplikasi yang dapat digabungkan dalam penelitian tersebut adalah dua media social yakni Facebook dan Gtalk dalam satu aplikasi. Teknologi yang digunakan adalah Java.

Usulan yang diajukan dalam skripsi ini adalah membuat chat room berbasis web dengan memanfaatkan teknologi websocket.

Dalam tinjauan pustaka dibawah ini terdapat lima refrensi sebagai berikut:

**Tabel 2. 1** Tabel Tinjauan Pustaka

Penulis Tahun	Judul Penelitian	Model	Interface
Andre (2015)	Aplikasi Panduan Kepramukaan Berbasis Android Menggunakan Format Pertukaran Data JSON	JSON (Java Script Object Notation)	GUI
Bakhtiar Wijayanto 2012	Merancang dan Membangun Aplikasi Chat Messenger Untuk Android	HTML, CSS, PHP dan MySQL.	GUI
Fitri Hardianto 2015	Aplikasi Grupchat Di Android Menggunakan Websocket	Java,XML, dan Websocket	GUI
Nicholas 2012	Pengembangan Aplikasi Chatroom Menggunakan Ajax Dan Php Berbasis Web	AJAX dan PHP	GUI
Fahsyani	Aplikasi Chat Engine	Java,	GUI

Ade Putra Wally 2014	Multi Sosial Media Berbasis Java	Facebook, GTalk	
Nisrina Luthfiati	Implementasi Websocket dan Node.js Dalam Aplikasi Chatting Berbasis Web	Websocket, Node.js ,dan mongodb	GUI

Sistem yang dibuat merupakan sistem berbasis web dimana pengguna harus melakukan *login* terlebih dahulu menggunakan *username* . Kemudian akan masuk ke halaman *chatting* untuk melakukan diskusi dengan mengirimkan sebuah pesan.

## 2. 2. Dasar Teori

### 2.2.1. Chatting

*Chat* adalah suatu *feature* atau suatu program di Internet untuk berkomunikasi langsung sesama pengguna internet yang sedang online / yang sedang sama-sama menggunakan Internet. Komunikasi ini dapat berupa teks (*text chat*) ataupun suara (*voice chat*). (Hardianto,Fitri. 2015)

### 2.2.2. WebSocket

*WebSocket* adalah standar baru untuk komunikasi *realtime* pada *Web* dan aplikasi *mobile*. *WebSocket* dirancang untuk diterapkan di *browser* web dan *server* web, tetapi dapat digunakan oleh aplikasi *client* atau *server*. *WebSocket* adalah protokol yang menyediakan saluran

komunikasi *full-duplex* melalui koneksi TCP tunggal. Protokol *WebSocket* sudah di standarisasi oleh IETF sebagai RFC 6455 pada tahun 2011, dan API *WebSocket* di Web IDL sedang distandarisasi oleh W3C. (Hardianto,Fitri. 2015)

### 2.2.3. Node.Js

Sebuah *platform* yang dirancang untuk pengembangan aplikasi *web* dan diimplementasikan di *server*. Node.js mampu melakukan proses baik secara *blocking (synchronous)* maupun *non-blocking (asynchronous)*. Dengan kemampuan Node.js yang dapat melakukan *non-blocking*, maka terbuka kemungkinan untuk membuat aplikasi *web instant messaging* yang ringan, *real-time* dan efisien. (nodejs.org)

### 2.2.4. JSON

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga bahasa perograman C (JSON.org)

### 2.2.5. MongoDB

MongoDB merupakan basis data noSQL (not only SQL) yang merupakan *document-oriented database* dan merupakan *open source project* yang tersedia di github (<https://github.com/mongodb/mongo>). (Julisman, Agung, 2015)

### 2.2.6. Socket Programming

Socket programming merupakan pemrograman ke arah protocol jaringan. Dalam pemrograman socket ini ada banyak protocol jaringan yang dapat diimplementasikan seperti UDC, HTTP, TCP, serta UDP.

Ilustrasi dari socket programming dapat dicontohkan seperti dua orang yang berkomunikasi dengan telepon. Konsep telepon di mana penerima dan penghubung akan melakukan komunikasi dua arah, ketika keduanya dapat terhubung. Yang menariknya konsep ini menggunakan nomor ekstension. Sehingga ketika penghubung menelpon nomor telepon yang dituju, ada nomor ekstensi yang harus ditekan, contohnya 114 untuk layanan keuangan dan 115 untuk layanan komunikasi.

Untuk pendefinisian istilah-istilah dalam socket programming dapat dirinci sebagai berikut :

- 1.) Server merupakan proses yang menjawab permintaan dari banyak client secara serempak. Jadi jika diilustrasikan, server ini

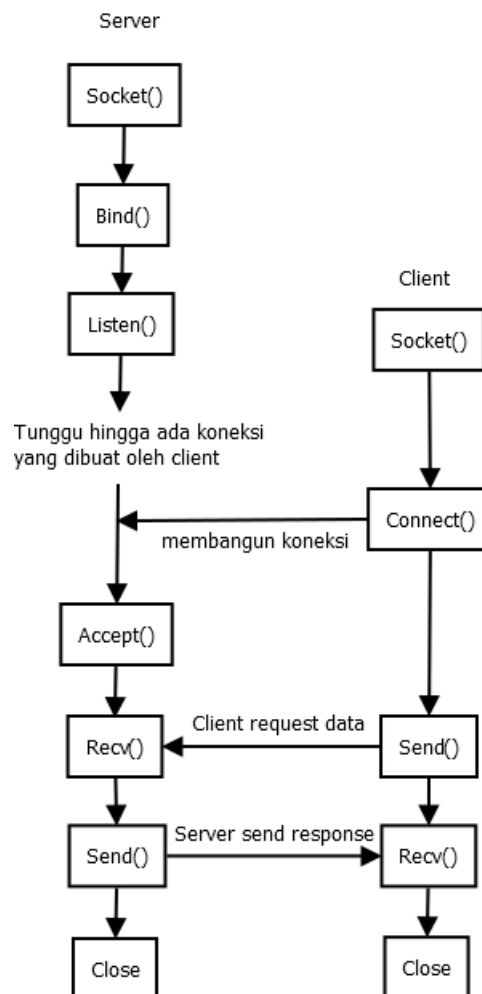
merupakan customer service yang menerima telepon dari banyak client.

- 2.) Client merupakan proses yang memulai suatu koneksi untuk melakukan hubungan komunikasi dengan server. Secara umum, akan ada banyak client yang mengadakan koneksi dengan suatu server dalam waktu yang sama. Jika diilustrasikan maka client merupakan orang yang menelpon customer service.
- 3.) IP Address merupakan alamat IP yang berbeda yang dimiliki oleh masing-masing komputer untuk dapat berkomunikasi dalam suatu jaringan. Jika diilustrasikan dalam socket programming maka alamat IP merupakan nomor telepon dari suatu instansi yang besar yang memiliki banyak layanan.
- 4.) Port Number merupakan nomor yang digunakan sebagai identifikasi atau layanan dari client yang ingin melakukan komunikasi. Jika bicara mengenai socket programming maka port number merupakan nomor ekstensi dari suatu nomor telepon. Sehingga ketika client ingin melakukan pembicaraan dengan server, client harus mengetahui IP address dan port untuk berkomunikasi. (Masya, Fajar., dan Fiade Andrew. 2015).

Analogi dari socket programming dapat digambarkan dalam beberapa fungsi. Fungsi-fungsi tersebut diantaranya adalah :

- 1.) Socket() digunakan untuk membangun *endpoint* dari komunikasi antara penghubung dengan customer service (client dan server).
- 2.) Bind() digunakan untuk menentukan nomor telepon beserta nomor-nomor ekstensi yang tersedia (IP address dan port number).
- 3.) Listen() digunakan ketika server atau customer service menunggu panggilan dari penghubung (server menunggu koneksi dari client).
- 4.) Connect() digunakan ketika penghubung menelpon customer service dan meminta layanan (client melakukan koneksi ke server).
- 5.) Accept() digunakan ketika customer service menerima telepon dan memberikan layanan pada penghubung (server menerima koneksi client).
- 6.) Send() dan Recv() digunakan untuk merepresentasikan proses komunikasi antara customer service dan penghubung (client dan server).
- 7.) Close() merupakan fungsi ketika client dan server sudah selesai berkomunikasi atau menutup telepon (client menghancurkan koneksi yang dibangun ke server).

Berikut ini adalah gambaran dalam bentuk diagram cara kerja dari socket server dan socket client dapat dilihat pada gambar 2.1.



**Gambar 2.1** Diagram cara kerja socket server dan socket client